

Deep Learning Jaringan Saraf Tiruan Untuk Pemecahan Masalah Deteksi Penyakit Daun Apel

Sutriawan¹⁾; Ahmad Zainul Fanani²⁾; Farrikh Alzami³⁾; Ruri Suko Basuki³⁾

¹⁾Program Studi Ilmu Komputer, Universitas Muhammadiyah Bima

^{1,2,3,4)}Program Studi Ilmu Komputer, Universitas Dian Nuswantoro

Sutriawan1612@gmail.com ¹⁾; a.zainul.fanani@dsn.dinus.ac.id; ²⁾;

Alzami@dsn.dinus.ac.id ³⁾; ruri.basuki@dsn.dinus.ac.id⁴⁾

ABSTRACT

Diseases on apple leaves are becoming a major issue for apple growers since they can cause the crop to fail. Due to the diversity of diseases that can affect apple leaves, it can be challenging for farmers to determine the cause of leaf damage. The purpose of this research is to evaluate a convolutional neural network (CNN) method for its potential use in solving the problem of apple leaf disease identification. Four types of illness are dealt with: normal, multi-illness, rusty, and scabby. Many methods, such as data preparation and a preset VGG-16 artificial neural network (CNN) architecture, are recommended for use in the deep artificial neural network processing method. The most precise outcomes occurred when the beta parameter value was set to $2 = 0.999$ at Epoch to 85/100 with an accuracy of 0.7582, and when the epsilon parameter value was set to $1e-07$ at Epoch to 32/100 with an accuracy of 0.7582 with the best accuracy.

Keywords: Deep Learning, Image, Convolutional Neural Network, VGG16

I. PENDAHULUAN

Selama beberapa tahun terakhir, penyakit daun apel telah menyebabkan kecemasan di kalangan petani apel. 4-6 minggu setelah pemangkasan ranting dan daun yang tidak produktif, gejala penyakit mulai terlihat pada daun. Dimulai dari daun tertua dan berlanjut ke daun termuda, daun awalnya mengembangkan bercak putih tidak menentu yang berwarna coklat dengan titik-titik hitam di permukaan atas, yang berlangsung hingga seluruh daun rontok. Teknik diagnosis konvensional, yang meliputi pemeriksaan pohon secara manual, mahal, tidak efektif, dan menantang. Untuk mengatasi masalah ini, ada peningkatan permintaan untuk sistem otomatis yang dapat dengan mudah mengidentifikasi penyakit pada tahap awal, melindungi tanaman dari kegagalan produksi, dan menghasilkan tanaman yang lebih berkualitas [1].

Model pembelajaran mendalam saat ini menjadi subjek dari berbagai penelitian yang melihat deteksi penyakit pada tanaman buah dan sayuran. Urat daun apel sangat rumit dan beragam, sehingga sulit untuk mengidentifikasi penyakit dengan nilai yang sebanding [2]. Karena hampir semua penyakit apel berdampak pada daun, maka daun merupakan sumber informasi yang sangat penting dan dapat diandalkan untuk identifikasi penyakit [3]. Namun, bergantung sepenuhnya pada sumber-sumber ini tidak mendorong

identifikasi penyakit yang cepat dan efektif dan juga dapat menyebabkan masalah lain yang terkait dengan subjektivitas. subjektif. Memantau kesehatan apel adalah proses yang memakan waktu, tetapi deteksi penyakit secara otomatis yang cepat selama produksi memungkinkan pemantauan yang jauh lebih menyeluruh dan akurat, dan juga membantu pemilik kebun membuat penilaian penyakit yang lebih terinformasi. Sangat penting untuk menerapkan tindakan pencegahan dan pengendalian penyakit secara tepat waktu untuk mencegah penyebaran penyakit guna memastikan pertumbuhan apel yang sehat dan memaksimalkan hasil panen [4]. Gambar 1 menunjukkan contoh data daun apel dengan kategori penyakit berbasis tekstur yang dapat dikenali berdasarkan bentuk, warna, dan jenis penyakit yang ada pada daun apel tersebut.

Sehat



Karat



Keropeng



Banyak penyakit



Gambar 1. Contoh variasi penyakit

Mendiagnosis penyakit tanaman bisa jadi merupakan hal yang menantang. Penyakit seperti keropeng, busuk hitam, dan karat cedar sangat berbahaya karena mereka dengan cepat menyebar ke bagian lain dari daun dan pada akhirnya membunuh tanaman jika tidak ditangani [5]. Ini adalah masalah serius ketika mencoba mendiagnosis berbagai macam penyakit, dan ini menyoroti perlunya mendasarkan diagnosis pada penampilan penyakit.

Selama investigasi ini membahas empat bentuk penyakit yang berbeda: sehat, banyak penyakit, berkarat, dan berkeropeng. Metode yang diusulkan dapat dibagi menjadi tiga langkah yang berbeda: mengidentifikasi area yang sakit, mengkategorikan area tersebut, dan melakukan analisis. Berikut ini adalah kontribusi utama yang diberikan oleh penelitian ini:

- Yang salah adalah praktik memisahkan saluran dan kemudian menyamakan setiap saluran secara terpisah. Tingkat intensitas gambar adalah yang disesuaikan oleh ekualisasi, bukan komponen warnanya. Oleh karena itu, penyetaraan histogram tidak bisa langsung diterapkan ke gambar yang hanya memiliki model warna RGB. Ini harus diterapkan sedemikian rupa sehingga level intensitasnya seimbang tanpa menghilangkan harmoni warna gambar agar efektif. Jadi, hal pertama yang perlu dilakukan yaitu mengubah ruang warna gambar dari RGB ke salah satu ruang warna yang membedakan antara nilai intensitas dan komponen warna.
- Dengan menetapkan ambang batas, sebuah nilai dipilih secara acak untuk digunakan sebagai ambang batas. Sebaliknya, metode Otsu tidak mengharuskan pengguna untuk memilih nilai, melainkan menghitung nilai secara otomatis.
- Setelah gambar histogram yang disamakan diperoleh dari tahap sebelum ini, metode segmentasi binarisasi Otsu harus digunakan untuk gambar tersebut.

Penelitian ini dibagi menjadi beberapa bab pembahasan, yang pertama berjudul "Pendahuluan" dan menjelaskan konteks historis serta tantangan penelitian yang terkait dengan deteksi penyakit pada daun apel. Bagian II, "Tinjauan pustaka", membahas penelitian terkait serta temuan penelitian sebelumnya. Bagian III: Metode penelitian,

diberikan gambaran umum mengenai sumber dan pemrosesan dataset, serta metode yang diusulkan untuk identifikasi penyakit daun apel. Bagian IV "Hasil dan Pembahasan" menjelaskan secara mendalam Hasil dari percobaan serta penilaian seberapa baik model yang diusulkan bekerja. Bagian V, "Kesimpulan", meringkaskan temuan penelitian, serta beberapa saran untuk pengembangan penelitian di masa depan.

II. TINJAUAN PUSTAKA

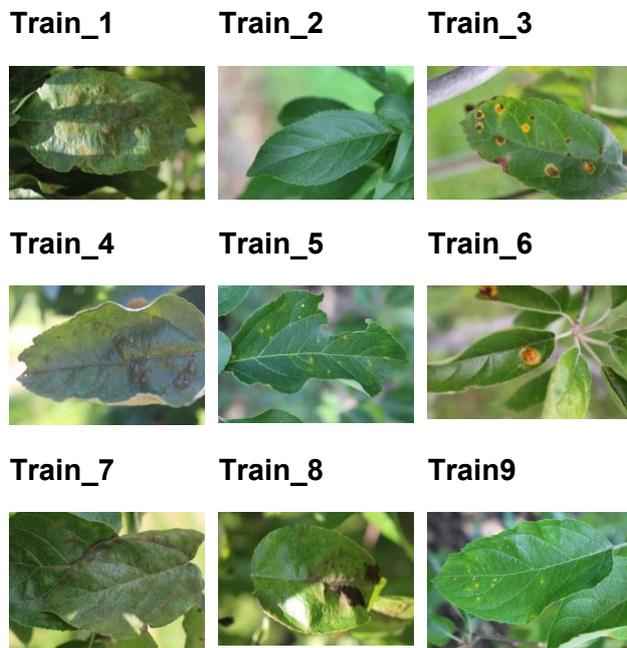
Mode pengembangan adaptif untuk deep learning yang dirancang khusus untuk mengidentifikasi penyakit pada daun tanaman, seperti Penggunaan segmentasi warna dalam hubungannya dengan transformasi wavelet diskrit yang dimodifikasi untuk diagnosis penyakit pada jaringan daun apel Deteksi penyakit daun apel berdasarkan jaringan syaraf tiruan *convolutional neural networks* (CNN) yang ditingkatkan Karena kerumitan dan variasi urat daun apel, serta sulitnya menentukan penyakit mana yang mirip satu sama lain, maka dikembangkanlah sebuah model pendeteksian dengan target penyakit daun apel yang baru dengan menggunakan teknik DF-Tiny-YOLO dan pendekatan deep learning. Model ini berhasil mengidentifikasi penyakit-penyakit [1,2]. Penelitian ini menggunakan tiga jenis model prediksi yang berbeda, yaitu CNN, SVM, dan KNN, untuk menganalisis foto-foto daun tanaman apel dan mengkategorikannya sebagai daun yang sehat atau sakit. Selain itu, sejumlah teknik pra-pemrosesan gambar digunakan untuk meningkatkan kualitas gambar dan menghilangkan noise yang mungkin berdampak pada hasil klasifikasi [5]. Identifikasi Penyakit Konvolusional Daun Apel yang Dimodelkan Menggunakan Jaringan Syaraf Tiruan (JST) menggunakan CNN bersama dengan model AlexNet untuk memberikan solusi terhadap masalah penyakit yang merusak daun pohon apel. Konvolusi digunakan dalam model untuk mengekstrak fitur kasar, yang membantu menjaga bidang reseptif yang besar sekaligus menurunkan jumlah parameter [6].

III. METODE PENELITIAN

Metode penelitian merupakan prosedur dan teknik penelitian. Antara satu penelitian dan penelitian lain, prosedur dan tekniknya akan berbeda. Kalau tidak berbeda berarti penelitian itu hanya mengulang penelitian yang telah ada sebelumnya.

3.1 Data Augmentation

Dataset daun Apple Public, yang diunduh dari repositori akses terbuka, digunakan di sini untuk analisis. Dataset ini berisi 1821 gambar daun Apple untuk tujuan pelatihan, dan 1821 gambar lainnya untuk tujuan pengujian.



Gambar 2. Contoh Data Pelatihan Penyakit Daun Apel

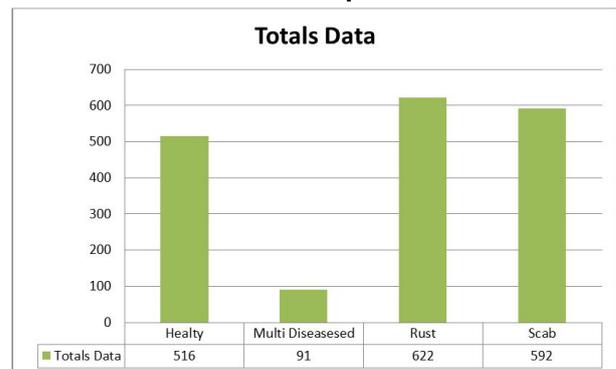
Pada training set, kita dapat melihat empat kategori penyakit yang berbeda (kesehatan, banyak penyakit, karat, dan kudis) seperti yang digambarkan pada Gambar 3.

	image_id	healthy	multiple_diseases	rust	scab
0	Train_0	0	0	0	1
1	Train_1	0	1	0	0
2	Train_2	1	0	0	0
3	Train_3	0	0	1	0
4	Train_4	1	0	0	0
...
1816	Train_1816	0	0	0	1
1817	Train_1817	1	0	0	0
1818	Train_1818	1	0	0	0
1819	Train_1819	0	0	1	0
1820	Train_1820	0	0	0	1

Gambar 3. Deskripsi kelas data pelatihan



Gambar 4. Contoh Data Pengujian Penyakit Daun Apel



Gambar 5. Grafik Persentase Kelas Penyakit Daun Apel

Gambar 3, yang mengilustrasikan persentase setiap kelas yang termasuk dalam kumpulan data, mengungkapkan informasi berikut:

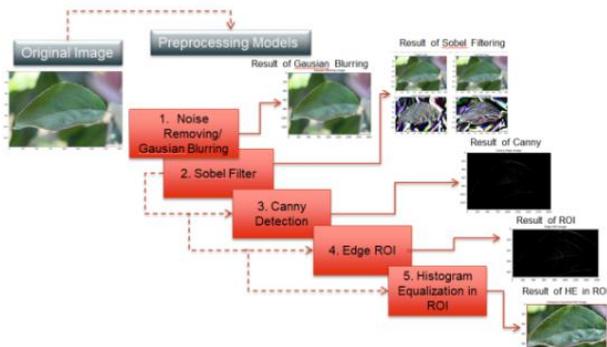
Tabel 1. Deskripsi Kelas Data

Sehat	516 gambar
Multi Penyakit	91 gambar
Karat	622 gambar
Keropeng	592 gambar

Dari data training dan testing yang masing-masing memiliki total 1821 total citra daun apel dengan total empat kelas penyakit, setiap citra daun memiliki beberapa jenis penyakit yang berbeda, antara lain sehat, banyak penyakit, karat, dan kudis. Jumlah total citra daun apel pada dataset pelatihan dan pengujian adalah 3621. Terdapat 516 sampel pada kelas Sehat, 91 sampel pada kelas multi-penyakit, 622 sampel karat, dan 592 sampel kudis. Hal ini menyebabkan dataset memiliki distribusi nilai yang tidak merata. Dataset yang tidak seimbang berpotensi menghasilkan hasil klasifikasi yang miring dan bias.

3.2 Preprocessing

Beberapa metode prapemrosesan yang digunakan dalam penelitian ini untuk memperbaiki gambar daun apel digambarkan dalam Gambar 6.



Gambar 6. Teknik Preprocessing yang Diusulkan

- Gaussian Blurring

Banyak teknik penghalusan gambar seperti Gaussian Blurring, Median Blurring, dll., sampai batas tertentu, bagus dalam menghilangkan sejumlah kecil noise. Dalam teknik-teknik tersebut, mengambil lingkungan kecil di sekitar piksel dan melakukan beberapa operasi seperti rata-rata tertimbang gaussian, nilai median, dll. untuk menggantikan elemen pusat.
- Sobel Filtering

Operator perbedaan orde pertama diskrit, operator Sobel dinamai sesuai dengan namanya. Fungsi kecerahan gambar dapat dihitung dengan bantuan operator ini. Hal ini diperlukan untuk melakukan perhitungan konvolusi matriks gambar dengan templat arah horizontal dan vertikal G_x dan G_y . Sebagai hasilnya, kita dapat memperoleh perkiraan kasar perbedaan kecerahan dalam bentuk gradien pada arah X dan Y [7].

Metode tradisional deteksi tepi sobel melibatkan operasi pada area piksel 3x3 menggunakan konvolusi templat yang sesuai, menyimpulkan nilai gradien piksel tengah, dan kemudian membandingkan nilai tersebut dengan nilai gradien ambang batas yang telah ditetapkan untuk menentukan piksel gambar yang sesuai dengan tepi gambar jika lebih besar dari kriteria ambang batas
- Canny Detection

Penggunaan pemfilteran Gaussian biasanya merupakan langkah awal dalam teknik identifikasi tepi cerdas apa pun. Dengan memperhalus gambar, noise dapat dihilangkan dengan menggunakan teknik

yang disebut pemfilteran Gaussian. Setelah itu, barulah dilakukan penghitungan gradien intensitas. Setelah eksekusi metode penekanan non-maksimum, muncullah penerapan proses double thresholding. Hal ini menjamin bahwa pendeteksian hanya terbatas pada bagian tepi yang kentara saja. Apabila berurusan dengan foto yang mengandung banyak noise, operasi Canny memberikan hasil yang bagus dari segi performa [7].

Implementasi algoritme pendeteksian Canny Edge dapat dibagi ke dalam lima langkah yang berbeda, yaitu sebagai berikut:

1. Penghalusan: Operasi yang dikenal sebagai penghalusan atau pemburaman dilakukan pada gambar untuk menghilangkan noise dari gambar.
2. Setelah gradien gambar ditemukan, tepi gambar hanya boleh ditandai di bagian yang memiliki magnitudo yang relatif tinggi.
3. Penekanan nilai non-maksimum Hanya nilai yang berada pada nilai maksimum lokalnya yang harus dianggap sebagai tepi.
4. Pengambilan ganda (double thresholding): Teknik ini digunakan untuk menentukan di mana calon tepi berada. Pelacakan tepi dengan histeresis: Setelah semua tepi yang tidak terhubung ke tepi yang sangat pasti atau kuat telah ditekan, tepi yang tersisa akan dipertimbangkan sebagai tepi akhir [9].

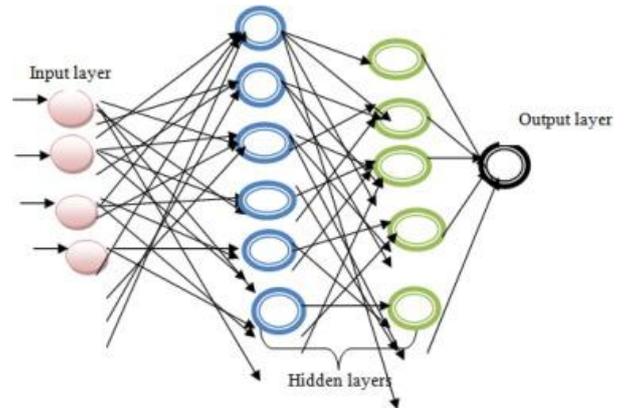
3.3. Deep Learning Modeling (DLM)

- Karakteristik perbandingan penyakit daun apel:

Tabel 2. Ciri-ciri daun yang sakit Karakteristik Daun Appel yang Sakit

Jenis penyakit	Karakteristik penyakit
Sehat	Sehat Memiliki bentuk elips atau lonjong, dan tepi daun bergerigi tumpul. Tidak terdapat bercak atau karat pada kedua sisi daun karena masih dalam tahap pertumbuhan.
Banyak penyakit	Multi penyakit Penyakit daun termasuk bercak merah, daun kosong, dan bentuk yang menyimpang adalah hal yang umum terjadi.
Karat	Karat Bercak-bercak berminyak pada awalnya berukuran kecil dan berwarna merah jingga,

Jenis penyakit	Karakteristik penyakit
	namun seiring pertumbuhannya, tampilannya berubah menjadi bulat, berwarna kuning jingga, dan bertepi merah.
Kopeng	Bintik-bintik penyakit keropeng cenderung berbentuk melingkar atau radial. Jamur dimulai dengan warna coklat kehijauan pada daun dan akhirnya menjadi hitam. Beberapa bercak penyakit dapat menyebar ke seluruh daun, sehingga terlihat hancur.



Gambar 7. Ilustrasi struktur deep learning

Gambar 7 menggambarkan konstruksi pengklasifikasi pembelajaran neural yang mendalam untuk analisis struktur-properti yang efektif. Lapisan input dan output, serta satu atau beberapa lapisan tersembunyi, hanyalah beberapa lapisan yang membentuk jaringan saraf. Neuron, yang berfungsi sebagai simpul pengaktifan, terhubung sepenuhnya dari satu lapisan ke lapisan berikutnya untuk membuat jaringan di setiap tingkat. Jumlah data yang dikumpulkan oleh lapisan input pada waktu t , yang direpresentasikan sebagai $Id(t)$, dikumpulkan pada saat ini. Neuron-neuron kemudian dihubungkan dari satu lapisan ke lapisan berikutnya dengan menggunakan bobot dinamis, yang dapat ditentukan dengan cara berikut:

$$Input(t) = \sum_{i=1}^n data_i * weight_i + bias \quad (1)$$

- Model training

Tabel 3, Perbandingan tipe Pengoptimal

Jenis pengoptimalan	Nama Parameter	Nilai Parameter
Adam	Adam	0.001
	beta_1	0.9
	beta_2	0.999
	epsilon	1e-07
	amsgrad	False
SGD	learning_rate	0.01
	momentum	0.0
	nesterov	False

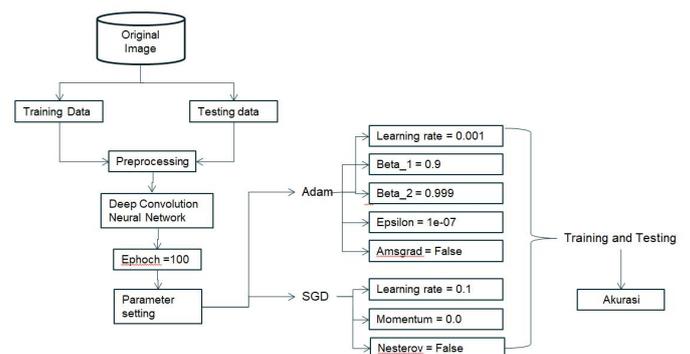
Ketika proses pelatihan dilakukan dengan membandingkan kinerja dua model pengoptimal yang berbeda dengan menggunakan beberapa parameter, setiap parameter diatur ke nilai default seperti yang ditunjukkan pada Tabel 3, dan Pengoptimal Adam dan Pengoptimal SGD adalah dua jenis pengoptimal yang digunakan dalam proses pelatihan.

3.4. Deep Convolution Neural Network

Deep learning adalah subbidang pembelajaran mesin yang mensimulasikan proses berpikir manusia dengan memproses informasi dengan cara yang mirip dengan cara manusia berpikir. Memproses informasi dan memahami wacana manusia dilakukan dengan menggunakan lapisan-lapisan dalam deep learning. Hubungan struktur-properti dari paduan teknik esensial dapat dimodelkan menggunakan teknik klasifikasi yang disebut deep neural learning, yang terdiri dari empat lapisan berbeda. Gambar 7 memberikan representasi visual dari struktur deep neural learning.

IV. HASIL DAN PEMBAHASAN

Pelatihan dan pengujian pada dataset daun apel memungkinkan untuk menilai kemampuan metode yang diusulkan. Gambar 8 menggambarkan model yang diusulkan.



Gambar 8. Model yang diusulkan Model yang Diusulkan

Semua hasil eksperimen diperoleh dengan membagi total waktu yang dihabiskan untuk

pelatihan dan pengujian sebesar 80:20 dan menerapkan faktor validasi sebesar 0,1. Gambar 9 memberikan ikhtisar ringkas tentang jumlah total gambar untuk setiap kelas, serta jumlah sampel yang digunakan dalam pengujian dan pelatihan.

```

VALIDATION_FACTOR = 0.1

val_size = int(len(x_train) * VALIDATION_FACTOR)

train_x = x_train[: len(x_train) - val_size]
train_y = y_train[: len(y_train) - val_size] # len(x_train) = len(y_train)

val_x = x_train[len(x_train) - val_size : len(x_train)]
val_y = y_train[len(y_train) - val_size : len(y_train)]

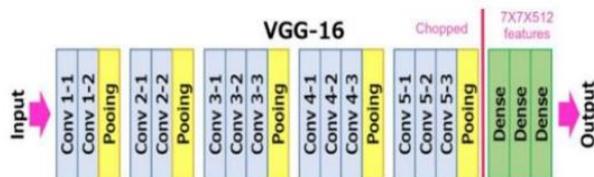
print("Shape of training data = ", train_x.shape, train_y.shape)
print("Shape of validation data = ", val_x.shape, val_y.shape)

Shape of training data = (1639, 224, 224, 3) (1639, 4)
Shape of validation data = (182, 224, 224, 3) (182, 4)
    
```

Gambar 9. Rincian pembagian data pelatihan

4.1. VGG-16 Arsitektur

Dalam pengujian ini, menggunakan arsitektur VGG16, yang merupakan arsitektur jaringan saraf tiruan deep convolutional dengan total 16 lapisan. ImageNet biasanya digunakan ketika mencari dataset gambar untuk digunakan dalam pelatihan arsitektur deep learning. ImageNet adalah dataset yang terdiri dari seribu kategori yang berbeda dan berisi jutaan foto. ImageNet sering digunakan dalam kompetisi yang melibatkan pembelajaran mesin. Hal pertama yang harus Anda lakukan adalah melakukan beberapa bentuk pembelajaran transfer. Untuk melakukan pembelajaran transfer, kita perlu menghapus lapisan yang paling baru, yaitu lapisan yang memiliki koneksi paling sedikit ke lapisan lain dalam model, dan kemudian memasang lapisan berikutnya. Tujuh lapisan yang berbeda membentuk arsitektur model VGG-16. Lapisan konvolusi membentuk lima lapisan, dan lapisan yang terhubung sepenuhnya membentuk dua lapisan [10].



Gambar 9. Arsitektur VGG-16 Arsitektur VGG-16

• Input Layer

Gambar mentah diambil pada lapisan input ini sebelum diteruskan ke lapisan berikutnya untuk diekstraksi fitur *convolutional* berikutnya.

Pemrosesan konvolusi diterapkan pada lapisan input. Pada lapisan ini, berbagai filter akan digunakan pada objek untuk menemukan fitur objek baru. Fitur atau properti ini digunakan dalam perhitungan apakah suatu proses pengujian cocok atau tidak.

$$Output = \frac{n + 2p - f}{s} + 1 \tag{2}$$

Di mana:

- n = panjang input atau tinggi input
- f = panjang filter kernel atau tinggi fitur kernel
- p = padding
- s = langkah

• Pooling

Lapisan penyatuan akan menerima fitur-fitur yang telah diekstrak dari data. Lapisan ini berguna untuk mengurangi dimensi objek gambar dengan ukuran besar dan mengurangi parameter objek gambar tersebut untuk menyimpan informasi penting. Lapisan ini bertanggung jawab untuk memastikan bahwa nilai maksimum dari setiap lapisan dipertahankan. Lapisan ini tidak memberikan kontribusi bobot pada total keseluruhan. Proses polling itu sendiri dipisahkan menjadi dua langkah yang berbeda: max-pooling, di mana sistem memilih hasil perhitungan konvolusi yang memiliki nilai terbesar, dan global average pooling, di mana sistem memilih nilai yang dirata-ratakan di semua hasil konvolusi.

$$Output_{maxpool} = \frac{n - f}{s} + 1 \tag{3}$$

• Fully Connected Layer

Lapisan paling atas adalah lapisan yang sepenuhnya terhubung, dan sistem beroperasi dengan terlebih dahulu mengubah gambar yang telah melalui tingkat penyaringan yang paling ketat ke dalam sistem pelabelan kategorikal. Pada lapisan ini, input akan diperoleh dari hasil pemrosesan yang dilakukan sebelumnya untuk menentukan apakah karakteristik saling berhubungan (berkorelasi) dengan kelas lainnya. Tujuan dari lapisan ini adalah untuk mengkonsolidasikan semua node individual dimensi ke dalam satu dimensi.

$$Z_j = \sum_{i=1}^c w_{ij}^T X_i + b_j \quad (4)$$

Dimana:

Z_j = Nilai keluaran dari jaringan

X_i = Nilai masukan dari ekstraksi fitur

W_{ij} = nilai bobot jaringan dengan ukuran $i \times j$

i = jumlah fitur masukan

j = jumlah kelas target

b_j = bias dari jaringan

Model: "vgg16"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
=====		
Total params: 14,714,688		
Trainable params: 14,714,688		
Non-trainable params: 0		

Gambar 10. Arsitektur Model VGG-19 Arsitektur Model VGG-19

Model: "sequential_25"

Layer (type)	Output Shape	Param #
dense_75 (Dense)	(None, 512)	262656
dropout_50 (Dropout)	(None, 512)	0
dense_76 (Dense)	(None, 512)	262656
dropout_51 (Dropout)	(None, 512)	0
dense_77 (Dense)	(None, 4)	2052
=====		
Total params: 527,364		
Trainable params: 527,364		
Non-trainable params: 0		

Gambar 11. Sequential model

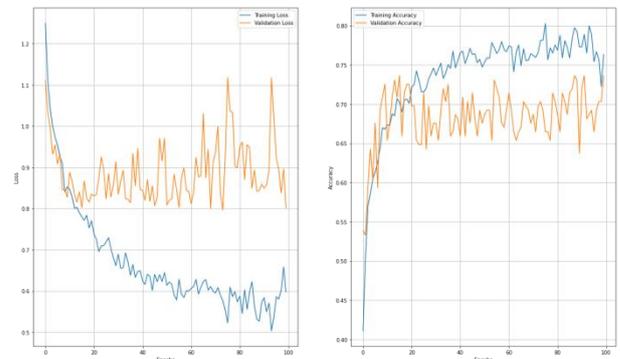
• Validasi dan Pengujian

Selama tahap validasi dan pengujian ini, faktor validasi diatur ke 0,1. Optimasi tipe dengan Adam dan SGD dilakukan dengan menggunakan pengaturan parameter Adam

berikut ini: learning rate = 0.1, beta 1 = 0.9, beta 2 = 0.999, epsilon = 1e-07, dan amsgrad = false. Optimasi dengan Adam menggunakan pengaturan parameter Adam berikut ini: learning rate = 0.1, momentum = 0.0, dan nesterov = false.

Tabel 3. Perbandingan akurasi pengujian model

Optimizers	Nama parameter	Nilai parameter	Ephoch	Akurasi
Adam	learning_rate	0.001	Epoch 86/100	0.7418
	beta_1	0.9	Epoch 55/100	0.7473
	beta_2	0.999	Epoch 85/100	0.7582
	epsilon	1e-07	Epoch 32/100	0.7582
	amsgrad	False	Epoch 86/100	0.7418
SGD	learning_rate	0.1	Epoch 81/100	0.6758
	momentum	0.0	Epoch 97/100	0.7088
	nesterov	False	Epoch 99/100	0.6813



Gambar 12. Loss validasi dan validasi akurasi

V. PENUTUP

Temuan dari penelitian ini telah memungkinkan kemajuan yang signifikan dalam diagnosis penyakit daun apel. Untuk mengidentifikasi kelainan pada tahap yang lebih dini, penelitian ini memanfaatkan teknik prapemrosesan yang menggabungkan sejumlah pendekatan metodologis. Apabila prosedur yang lebih konvensional tampaknya membentur tembok, metode prapemrosesan dapat meningkatkan kualitas gambar dengan menggunakan teknik seperti pemerataan histogram, deteksi tepi, dan lain-lain. Hal ini berdampak pada peningkatan efisiensi pembelajaran mendalam yang dilakukan. Setelah menerapkan model yang diusulkan, kemudian beralih ke sepasang model pengoptimalan yang dikenal sebagai Adam

dan SGD. Nilai parameter untuk setiap model optimasi berbeda satu sama lain. Hasil dari pengujian ini dianalisis untuk membuat perbandingan tentang efektivitas berbagai model optimasi. Tabel 3 menyajikan hasil eksperimen deep learning yang dilakukan dengan berbagai model optimasi, dengan menggunakan convolutional neural network (CNN). Ketika menggunakan model optimasi Adam dengan nilai parameter $\beta_2 = 0.999$ pada Epoch ke 85/100 dengan akurasi 0.7582, ketika menggunakan nilai parameter epsilon $1E-07$ pada Epoch ke 32/100 dengan akurasi 0.7582 yang dapat mencapai tingkat presisi tertinggi. Karena tingginya tingkat akurasi yang dicapai, jelaslah bahwa masalah deteksi penyakit pada daun apel berhasil ditangani.

DAFTAR PUSTAKA

- [1] S. Hasan, S. Jahan, and M. I. Islam, "Disease detection of apple leaf with combination of color segmentation and modified DWT," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 9, pp. 7212–7224, 2022, doi: 10.1016/j.jksuci.2022.07.004.
- [2] J. Di and Q. Li, "A method of detecting apple leaf diseases based on improved convolutional neural network," *PLoS One*, vol. 17, no. 2 February, pp. 1–15, 2022, doi: 10.1371/journal.pone.0262629.
- [3] A. R. Ram, "Plant Disease Detection Using Leaf Pattern : A Review," vol. 2, no. 6, pp. 774–780, 2015.
- [4] M. A. Khan *et al.*, "CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features," *Comput. Electron. Agric.*, vol. 155, no. February, pp. 220–236, 2018, doi: 10.1016/j.compag.2018.10.013.
- [5] S. Alqethami, B. Almtanni, W. Alzhrani, and M. Alghamdi, "Disease Detection in Apple Leaves Using Image Processing Techniques," *Eng. Technol. Appl. Sci. Res.*, vol. 12, no. 2, pp. 8335–8341, 2022, doi: 10.48084/etasr.4721.
- [6] L. Fu, S. Li, Y. Sun, Y. Mu, T. Hu, and H. Gong, "Lightweight-Convolutional Neural Network for Apple Leaf Disease Identification," *Front. Plant Sci.*, vol. 13, no. May, pp. 1–10, 2022, doi: 10.3389/fpls.2022.831219.
- [7] R. A. A S and S. Gopalan, "Comparative Analysis of Eight Direction Sobel Edge Detection Algorithm for Brain Tumor MRI Images," *Procedia Comput. Sci.*, vol. 201, no. C, pp. 487–494, 2022, doi: 10.1016/j.procs.2022.03.063.
- [8] K. Zhang, Y. Zhang, P. Wang, Y. Tian, and J. Yang, "An improved sobel edge algorithm and FPGA implementation," *Procedia Comput. Sci.*, vol. 131, pp. 243–248, 2018, doi: 10.1016/j.procs.2018.04.209.
- [9] S. M. Abid Hasan and K. Ko, "Depth edge detection by image-based smoothing and morphological operations," *J. Comput. Des. Eng.*, vol. 3, no. 3, pp. 191–197, 2016, doi: 10.1016/j.jcde.2016.02.002.
- [10] T. Sulistyowati, F. Al Zami, and R. A. Pramunendar, "VGG16 Deep Learning Architecture Using Imbalance Data Methods For The Detection Of Apple Leaf Diseases," *Monet. J. Keuang. Dan Perbank.*, vol. 11, no. 1, pp. 41–53, 2023.